

# Implementasi Sistem Persamaan Linear untuk Prediksi Pola Recoil Senjata dalam PUBG Mobile

Danendra Shafi Athallah - 13523136<sup>1</sup>

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

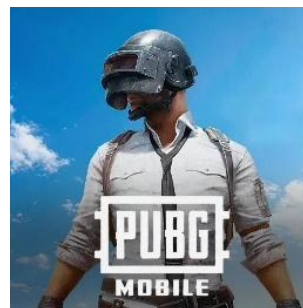
<sup>1</sup>danendra1967@gmail.com, 13523136@std.stei.itb.ac.id

**Abstract**—Penelitian ini menyajikan implementasi sistem persamaan linear untuk memprediksi pola recoil senjata dalam PUBG Mobile, sebuah game battle royale yang sangat populer. Recoil, yang ditandai oleh pergerakan vertikal dan horizontal crosshair senjata selama tembakan berkelanjutan, secara signifikan mempengaruhi performa pemain dan keseimbangan permainan. Pemodelan pola recoil dari tiga jenis senapan serbu yang umum digunakan yaitu AKM, M416, dan SCAR-L menggunakan persamaan linear yang mempertimbangkan variabel seperti waktu ( $t$ ), jumlah peluru yang ditembakkan ( $n$ ), serta parameter spesifik senjata ( $p$  dan  $r$ ). Data dikumpulkan melalui pengujian dalam game yang terkontrol, dan metode eliminasi Gauss dikombinasikan dengan metode least squares digunakan untuk menentukan koefisien model linear. Hasil penelitian menunjukkan bahwa SCAR-L memiliki recoil paling stabil dengan Root Mean Squared Error (RMSE) terendah, diikuti oleh M416 dan AKM yang menunjukkan tingkat ketidakpastian yang lebih tinggi. Pendekatan pemodelan ini menyediakan kerangka matematis untuk memahami dan mengoptimalkan kontrol recoil dalam PUBG Mobile, memberikan wawasan bagi pemain yang ingin meningkatkan gameplay mereka dan pengembang yang bertujuan untuk memperbaiki keseimbangan senjata.

**Keywords**—sistem persamaan linear, PUBG Mobile, recoil pattern, assault rifle.

## I. PENDAHULUAN

*PlayerUnknown's Battlegrounds Mobile* (PUBG Mobile) merupakan salah satu *game battle royale* yang paling populer di dunia dengan lebih dari 500 juta kali unduhan secara global. Dalam game ini, kemampuan mengendalikan senjata, khususnya dalam aspek *recoil control*, menjadi faktor krusial yang menentukan performa pemain. *Recoil* atau hentakan mundur senjata adalah fenomena yang terjadi ketika senjata ditembakkan secara terus-menerus, menghasilkan pola pergerakan yang dapat diprediksi namun sulit untuk dikontrol secara konsisten.



Gambar 1. PUBG Mobile

(Sumber:

<https://play.google.com/store/apps/details?id=com.tencent.ig&hl=id&pli=1> )

Dalam konteks matematika, pola *recoil* senjata dapat dimodelkan sebagai sebuah sistem persamaan linear yang merepresentasikan pergerakan *vertical* dan *horizontal* dari *crosshair* senjata. Pemahaman dan prediksi pola ini memiliki implikasi signifikan, tidak hanya dalam pengembangan kemampuan pemain tetapi juga dalam aspek *game design* dan *balancing*.

Penelitian ini berfokus pada implementasi sistem persamaan linear untuk memprediksi dan menganalisis pola recoil tiga senjata bertipe *assault rifle* (AR) yang umum digunakan dalam game PUBG Mobile: AKM, M416, dan SCAR-L. Ketiga senjata ini dipilih karena merepresentasikan spektrum *recoil* yang berbeda (tinggi, medium, rendah) dan memiliki karakteristik yang *distinct*. Dengan menggunakan pendekatan matematis, penelitian ini bertujuan untuk mengidentifikasi pola *recoil* masing-masing senjata akan dilakukan dengan pendekatan model matematis berbasis sistem persamaan linear untuk memprediksi pola *recoil*. Penelitian ini menggunakan versi terbaru dalam PUBG Mobile yakni versi 3.5 per tanggal 11 November pukul 2:00 (UTC+0).

Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pemahaman mekanika *shooting game mobile* dari perspektif matematis, serta menyediakan framework untuk analisis serupa pada game-game sejenis. Selain itu, model yang dikembangkan dapat membantu pemain dalam memahami dan mengoptimalkan kontrol *recoil*, serta memberikan *insight* bagi pengembang *game* dalam proses *weapon balancing*.

## II. DASAR TEORI

### A. Sistem Persamaan Linier

Sistem persamaan linear (SPL) merupakan himpunan persamaan linear yang saling berkaitan dan harus dipenuhi secara bersamaan. Dalam konteks matematika, SPL dengan  $m$  persamaan dan  $n$  variabel dapat dituliskan dalam bentuk standar:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m \end{cases}$$

Dengan :

- $a_{ij}$  adalah koefisien variabel ke- $j$  pada persamaan ke- $i$ .
- $x_j$  adalah variabel ke- $j$  yang nilainya belum diketahui.
- $b_i$  adalah konstanta pada persamaan ke- $i$ .
- $m$  adalah jumlah persamaan.
- $n$  adalah jumlah variabel.

### B. Representasi Matriks dan Analisis Solusi

SPL dapat direpresentasikan dalam bentuk matriks augmented  $[A|b]$  untuk memudahkan analisis dengan bentuk umum sebagai berikut:

$$\left[ \begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_m \end{array} \right]$$

Berdasarkan teori aljabar linear, analisis solusi SPL menghasilkan tiga kemungkinan utama. Pertama, Solusi Unik (Tunggal) terjadi ketika sistem memiliki tepat satu solusi. Hal ini terjadi apabila matriks koefisien memiliki *rank* penuh, yang berarti bahwa jumlah persamaan ( $m$ ) sama dengan jumlah variabel ( $n$ ) dan determinan matriks koefisien tidak sama dengan nol. Kondisi ini memastikan bahwa setiap persamaan dalam sistem memberikan informasi yang diperlukan untuk menentukan nilai unik dari setiap variabel.

Kedua, Solusi Banyak (Tak Hingga) muncul ketika sistem memiliki tak hingga banyak solusi. Situasi ini terjadi apabila *rank* matriks koefisien kurang dari jumlah variabel, yang menyebabkan adanya variabel bebas dalam solusi. Dalam kasus ini, solusi tidak dapat ditentukan secara unik dan dinyatakan dalam bentuk parametrik, memungkinkan variasi nilai variabel bebas untuk menghasilkan berbagai solusi yang valid.

Ketiga, Tidak Ada Solusi terjadi ketika sistem tidak memiliki solusi yang memenuhi semua persamaan secara simultan. Kondisi ini biasanya disebabkan oleh adanya kontradiksi dalam sistem, di mana *rank* matriks augmented lebih besar daripada *rank* matriks koefisien. Kontradiksi ini menandakan bahwa tidak ada nilai variabel yang dapat memenuhi seluruh persamaan sekaligus, sehingga sistem dinyatakan tidak konsisten dan tidak memiliki solusi.

### C. Metode Eliminasi Gauss

Metode Eliminasi Gauss merupakan algoritma fundamental untuk menyelesaikan sistem persamaan linear dengan mengubah matriks augmented menjadi bentuk eselon baris. Proses ini dilakukan melalui serangkaian Operasi Baris Elementer (OBE) yang terdiri dari tiga jenis operasi dasar. Operasi pertama adalah perkalian baris dengan skalar tidak nol ( $k_{(i)}R_{(i)}$ ), yang memungkinkan penyekalaan nilai-nilai dalam satu baris. Operasi kedua adalah pertukaran posisi dua baris ( $R_{(i)} \leftrightarrow R_{(j)}$ ), yang berguna untuk mengatur posisi pivot. Operasi ketiga adalah penambahan kelipatan suatu baris ke baris lain ( $R_{(i)} + kR_{(j)}$ ), yang digunakan untuk mengeliminasi nilai-nilai di bawah pivot.

Implementasi metode Eliminasi Gauss dimulai dengan pembentukan matriks augmented dari SPL yang akan diselesaikan. Selanjutnya, matriks tersebut ditransformasikan ke dalam bentuk eselon baris melalui serangkaian langkah sistematis. Langkah pertama adalah mencari pivot (elemen pertama tidak nol) di kolom pertama. Setelah pivot ditemukan, semua elemen di bawah pivot dibuat bernilai nol menggunakan operasi baris elementer. Proses ini kemudian dilanjutkan ke kolom berikutnya dan diulangi hingga tercapai bentuk eselon baris. Setelah bentuk eselon baris diperoleh, nilai variabel-variabel dapat ditemukan menggunakan teknik substitusi mundur.

Metode Eliminasi Gauss memiliki beberapa keunggulan yang membuatnya menjadi pilihan utama dalam penyelesaian sistem persamaan linear. Pertama, metode ini dapat diimplementasikan secara efisien dalam komputasi, membuatnya ideal untuk penyelesaian sistem dengan ukuran besar. Kedua, metode ini memiliki kemampuan untuk mendeteksi inkonsistensi dalam sistem, memberikan informasi penting tentang keberadaan solusi. Terakhir, proses eliminasi memberikan informasi tentang *rank* matriks, yang berguna untuk analisis lebih lanjut.

### D. Pemodelan Sistem Recoil

Dalam konteks PUBG Mobile, *recoil* merupakan fenomena kompleks yang dapat dimodelkan menggunakan kombinasi SPL dan analisis pola. Pemodelan ini mencakup beberapa aspek penting:

#### 1. Komponen Recoil

*Recoil* Vertikal (V) adalah pergerakan senjata ke arah atas saat ditembakkan. Pola *recoil* vertikal umumnya lebih prediktif karena dipengaruhi oleh karakteristik senjata, seperti jenis dan kekuatan *recoil* dasar. Pemain dapat memanfaatkan pola ini untuk mengendalikan tembakan dengan lebih baik dan meningkatkan akurasi. *Recoil* Horizontal (H) menggambarkan pergerakan senjata ke kiri atau kanan selama penembakan. *Recoil* horizontal memiliki elemen acak yang lebih tinggi, membuatnya kurang prediktif dan lebih menantang untuk dikontrol. Variasi pola *recoil* horizontal sangat bergantung pada tipe senjata, yang memerlukan adaptasi teknik pengendalian oleh pemain.

## 2. Parameter Pemodelan

Dalam pemodelan sistem *recoil*, beberapa parameter kunci digunakan untuk merepresentasikan berbagai aspek pengendalian tembakan. Parameter-parameter tersebut meliputi:

- Waktu ( $t$ ): Representasi waktu sejak tembakan pertama.
- Jumlah peluru ( $n$ ): Urutan peluru dalam mode *auto fire*.
- *Base recoil* ( $b$ ): Karakteristik dasar senjata.
- *Recoil pattern* ( $p$ ): Pola pergerakan spesifik.
- *Random factor* ( $r$ ): Elemen ketidakpastian.

## 3. Sistem Persamaan

Berdasarkan parameter dan komponen yang diperlukan, berikut merupakan model SPL dengan formula berikut:

$$\begin{cases} V(t) = \alpha_1 t + \alpha_2 n + \alpha_3 p + \varepsilon_1 \\ H(t) = \beta_1 t + \beta_2 n + \beta_3 r + \varepsilon_2 \end{cases}$$

Dimana:

- $V(t)$  adalah pergerakan vertikal senjata pada waktu  $t$ .
- $H(t)$  adalah pergerakan horizontal senjata pada waktu  $t$ .
- $\alpha_1, \alpha_2, \alpha_3$  adalah koefisien yang merepresentasikan kontribusi waktu  $t$ , jumlah peluru  $n$ , dan pola *recoil*  $p$  terhadap pergerakan vertikal.
- $\beta_1, \beta_2, \beta_3$  adalah koefisien yang merepresentasikan kontribusi waktu  $t$ , indeks peluru  $n$ , dan faktor acak  $r$  terhadap pergerakan horizontal.
- $\varepsilon_1, \varepsilon_2$  adalah *error term* untuk variasi *random*.

Pada persamaan  $V(t)$  koefisien  $\alpha_1, \alpha_2, \alpha_3$  adalah koefisien yang mengalikan variabel waktu ( $t$ ), jumlah peluru ( $n$ ), dan pola *recoil* ( $p$ ) terhadap pergerakan vertikal senjata. Nilai koefisien ini dapat bernilai positif atau negatif dengan kondisi:

- Positif ( $\alpha_i > 0$ ): Menunjukkan pergerakan *recoil* ke atas.
- Negatif ( $\alpha_i < 0$ ): Menunjukkan pergerakan *recoil* ke bawah.

Sedangkan pada persamaan  $H(t)$  koefisien  $\beta_1, \beta_2, \beta_3$  juga dapat bernilai positif atau negatif:

- Positif ( $\beta_i > 0$ ): Menunjukkan pergerakan *recoil* ke kanan.
- Negatif ( $\beta_i < 0$ ): Menunjukkan pergerakan *recoil* ke kiri.

## E. Karakteristik Assault Rifles

Sebelum membahas spesifikasi *detail* dari masing-masing senjata Assault Rifle, penting untuk memahami beberapa istilah spesifikasi yang relevan:

- **Daya (Damage):** Jumlah kerusakan yang diberikan oleh senjata kepada musuh setiap kali tembakan mengenai target. Semakin tinggi nilai daya, semakin besar potensi senjata untuk mengeliminasi musuh dengan cepat.
- **Laju Tembakan:** Kecepatan senjata dalam menembakkan peluru, biasanya diukur dalam detik per peluru. Senjata dengan laju tembakan tinggi memungkinkan penembakan yang lebih cepat dan

kontinu.

- **Recoil:** Hentakan atau pergerakan mundur senjata saat ditembakkan. *Recoil* mempengaruhi stabilitas tembakan dan akurasi, sehingga kontrol *recoil* yang baik sangat penting untuk menjaga presisi dalam menembak.
- **Jangkauan:** Seberapa jauh peluru senjata dapat mencapai target dengan efektif. Senjata dengan jangkauan panjang cocok untuk pertempuran jarak jauh, sementara jangkauan pendek lebih efektif di medan pertempuran dekat.
- **Kecepatan Isi Peluru:** Kecepatan senjata dalam mengisi ulang magazine setelah kosong. Kecepatan reload yang tinggi memungkinkan pemain untuk kembali ke pertempuran dengan cepat tanpa banyak jeda.

Dalam penelitian ini, fokus diberikan pada tiga jenis Assault Rifle dengan karakteristik berbeda:

### 1. AKM



Gambar 2. AKM

(Sumber:

[https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/arms\\_content.shtml?1#2020031506](https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/arms_content.shtml?1#2020031506))

AKM adalah salah satu *Assault Rifle* unggulan dalam PUBG Mobile yang memiliki Daya (Damage) sebesar 48, memungkinkan senjata ini memberikan kerusakan signifikan pada setiap tembakan yang mengenai target. Dengan Laju Tembakan sebesar 84 detik per peluru, AKM mampu menembakkan peluru dengan cepat yang dapat memberikan tekanan pada musuh. *Recoil* pada AKM tergolong berat yakni sebesar 76 sehingga membutuhkan kontrol *recoil* yang baik untuk menjaga akurasi tembakan. Kecepatan Isi Peluru AKM adalah 76 per *magazine* peluru, memberikan keseimbangan antara kapasitas amunisi dan frekuensi reload yang dapat diandalkan dalam pertempuran intens. Karakteristik ini menjadikan AKM pilihan populer bagi pemain yang mengutamakan daya hancur tinggi namun memiliki *recoil* yang sulit untuk dikontrol disebabkan oleh *attachment* yang terbatas dan daya yang besar.

## 2. M416



Gambar 3. M416

(Sumber:

[https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/arms\\_content.shtml?1#2020031501](https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/arms_content.shtml?1#2020031501))

M416 adalah salah satu *Assault Rifle* terbaik dalam PUBG Mobile yang paling diminati hingga dalam PMGC (PUBG Mobile Global Championship) yang disebabkan oleh Laju Tembakan sebesar 85 detik per peluru, M416 mampu menembakkan peluru dengan cepat yang dapat memberikan tekanan pada musuh. Dengan Daya sebesar 41, memungkinkan senjata ini memberikan kerusakan signifikan pada setiap tembakan yang mengenai target. *Recoil* pada M416 tergolong baik apabila seluruh *attachment* seperti pada komponen rekomendasi telah dipasang, dengan nilai *recoil* sebesar 76 sehingga membutuhkan kontrol *recoil* yang baik untuk menjaga akurasi tembakan. Kecepatan Isi Peluru M416 adalah 76 dengan kapasitas 30 peluru, memberikan keseimbangan antara kapasitas amunisi dan frekuensi *reload* yang dapat diandalkan dalam pertempuran intens.

## 3. SCAR-L



Gambar 4. SCAR-L

(Sumber:

[https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/arms\\_content.shtml?1#2020031503](https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/arms_content.shtml?1#2020031503))

SCAR-L adalah salah satu *Assault Rifle* dalam PUBG Mobile yang walaupun memiliki Daya lebih kecil dibandingkan dengan AKM dan M416, SCAR-L dapat memberikan kerusakan yang lebih rendah pada setiap tembakan. Meskipun demikian, SCAR-L menawarkan *Recoil* yang sedikit lebih ringan tanpa perlu menggunakan *attachment*, memudahkan pemain dalam mengendalikan tembakan dan menjaga akurasi. Karakteristik ini menjadikan SCAR-L pilihan ideal bagi pemain yang mengutamakan kestabilan dan presisi dalam pertempuran jarak menengah di awal pertandingan.

Meskipun perbedaan antara SCAR-L, AKM, dan M416 terlihat kecil pada spesifikasi dasar, perbedaan ini berdampak signifikan pada kontrol *recoil*, laju tembakan dan akurasi tembakan. Setiap senjata memiliki karakteristik unik yang mempengaruhi stabilitas dan efektivitas dalam berbagai situasi pertempuran. Selain itu, variasi preferensi dan gaya bermain pemain menuntut analisis mendalam terhadap masing-masing senjata untuk menghasilkan model prediksi yang akurat. Dengan demikian, detail perbedaan tersebut sangat relevan dan penting untuk memahami mekanika *recoil*.

## III. IMPLEMENTASI

### A. Pengumpulan Data

Sebelum melakukan analisis sistem persamaan linear, data pola *recoil* perlu diekstraksi dari PUBG Mobile menggunakan *screen recorder* dan pengolahan data. Diawali dengan *setup* spesifikasi serta pengaturan *in-game* yang sudah di standarisasi guna data *recoil pattern* yang konsisten sebagai berikut:

- Graphic: HD
- Frame Rate: Extreme
- Style: Classic
- Sensitivity: High

Pengambilan data dilakukan dengan prosedur posisi berdiri, target *dummy* pada jarak 10m, tidak menggunakan *attachments* pada senjata, ADS dengan *scope* tipe *Red Dot Sight*, tanpa menahan *recoil*, dan *gyroscope* dinonaktifkan.



Gambar 5. Visualisasi Titik Impact Peluru pada Target dengan SCAR-L

(Sumber: Dokumentasi Pribadi)



**Gambar 6.** Visualisasi Titik Impact Peluru pada Target dengan M416  
(Sumber: Dokumentasi Pribadi)



**Gambar 7.** Visualisasi Titik Impact Peluru pada Target dengan AKM  
(Sumber: Dokumentasi Pribadi)

Data rekaman pola tembakan dilakukan sebanyak 30 peluru per senjata untuk AKM, M416, dan SCAR-L. Data mentah mencakup waktu, koordinat tembakan ( $x,y$ ) dan kecepatan vertikal ( $V$ ) serta horizontal ( $H$ ).

### B. Pemrosesan Data

Proses data mentah dilakukan melalui beberapa tahapan berikut:

```

4 def process_recoil_data(csv_path):
5     try:
6         # baca data CSV
7         print(f"Reading file: {csv_path}")
8         df = pd.read_csv(csv_path)
9         print(f"Original data shape: {df.shape}")
10        # normalisasi koordinat relatif terhadap tembakan pertama
11        first_x = df.iloc[0]['x_coord']
12        first_y = df.iloc[0]['y_coord']
13        first_time = df.iloc[0]['time']
14        df['x_normalized'] = df['x_coord'] - first_x
15        df['y_normalized'] = df['y_coord'] - first_y
16        df['time_normalized'] = df['time'] - first_time
17        # hitung delta pergerakan
18        df['delta_x'] = df['x_normalized'].diff().fillna(0)
19        df['delta_y'] = df['y_normalized'].diff().fillna(0)
20        df['delta_time'] = df['time_normalized'].diff().fillna(0.001)
21        # hitung V dan H
22        df['V'] = (df['delta_y'] / df['delta_time']).replace([np.inf, -np.inf], np.nan)
23        df['H'] = (df['delta_x'] / df['delta_time']).replace([np.inf, -np.inf], np.nan)
24        # filter outliers (hanya jika ada data yang valid)
25        if not df['V'].isna().all() and not df['H'].isna().all():
26            Q1_v = df['V'].quantile(0.25)
27            Q3_v = df['V'].quantile(0.75)
28            IQR_v = Q3_v - Q1_v
29            Q1_h = df['H'].quantile(0.25)
30            Q3_h = df['H'].quantile(0.75)
31            IQR_h = Q3_h - Q1_h
32            df = df[
33                (df['V'] >= Q1_v - 1.5*IQR_v) & (df['V'] <= Q3_v + 1.5*IQR_v) &
34                (df['H'] >= Q1_h - 1.5*IQR_h) & (df['H'] <= Q3_h + 1.5*IQR_h)
35            ]
36        # simpan hasil processing
37        output_path = csv_path.replace('.csv', '_processed.csv')
38        df.to_csv(output_path, index=False)
39        print(f"Processed data saved to {output_path}")
40        print(f"Processed data shape: {df.shape}")
41        return df
42    except Exception as e:
43        print(f"Error processing file {csv_path}: {str(e)}")
44        return None

```

**Gambar 8.** Source code pemrosesan data  
(Sumber: Dokumentasi Pribadi)

Data tembakan dinormalisasi relatif terhadap titik awal tembakan. Hal ini memastikan pola *recoil* bersifat relatif dan

memudahkan analisis model matematis. Selanjutnya pada perubahan posisi koordinat ( $\Delta x, \Delta y$ ) dan waktu ( $\Delta t$ ) dihitung untuk setiap tembakan. Proses ini penting untuk menganalisis pergerakan pola *recoil* setiap peluru. Untuk memastikan data konsisten, *outlier* dihilangkan menggunakan metode Interquartile Range (IQR). *Outlier* yang berada di luar rentang 1,5 kali IQR dikeluarkan.

### C. Model Matematis

Setelah data diproses, model matematis dikembangkan menggunakan sistem persamaan linear berikut:

$$\begin{cases} V(t) = \alpha_1 t + \alpha_2 n + \alpha_3 p + \epsilon_1 \\ H(t) = \beta_1 t + \beta_2 n + \beta_3 r + \epsilon_2 \end{cases}$$

Model ini diimplementasikan dalam Python menggunakan matriks augmented  $A$  dan vektor target  $b$ . Matriks  $A$  mencakup waktu ( $t$ ), indeks peluru ( $n$ ), dan parameter *recoil* ( $p$  dan  $r$ ) yang spesifik untuk senjata masing-masing senjata, sedangkan  $b$  adalah vektor hasil observasi ( $V$  untuk dimensi vertikal dan  $H$  untuk dimensi horizontal). Penyelesaian model dilakukan menggunakan metode *least squares*, yang memanfaatkan pendekatan eliminasi Gauss untuk menghitung solusi optimal koefisien model ( $\alpha_1, \alpha_2, \alpha_3$  dan  $\beta_1, \beta_2, \beta_3$ ).

Metode *least squares* menyelesaikan sistem persamaan dengan cara meminimalkan galat kuadrat antara data aktual dan prediksi model. Dalam Python, implementasi ini dilakukan dengan memanfaatkan fungsi `linalg.lstsq` dari pustaka SciPy. Fungsi ini secara otomatis menghitung  $(A^T A)x = A^T b$ , dimana  $x$  adalah koefisien model yang diestimasi. Penyelesaian dimulai dengan membangun matriks  $A$ , yang disusun sebagai kombinasi komponen model SPL  $V(t)$  dan  $H(t)$ . Berikut adalah langkah-langkah penyelesaian dalam kode Python:

```

5 def fit_recoil_model(processed_csv_path, weapon_params):
6     """
7     Fit SPL model untuk data recoil
8     """
9     # baca data yang udah diproses
10    df = pd.read_csv(processed_csv_path)
11    # setup matriks untuk SPL
12    t = df['time_normalized'].values
13    n = np.arange(len(t)) # jumlah peluru
14    # setup parameter spesifik senjata
15    p = np.full_like(t, weapon_params['recoil_pattern']) # nilai pola recoil
16    r = np.full_like(t, weapon_params['random_factor']) # nilai random spread
17    # matriks A untuk vertical dan horizontal
18    A = np.column_stack([t, n, p])
19    # vektor b untuk vertical dan horizontal
20    b_v = df['V'].values
21    b_h = df['H'].values
22    # hasil SPL menggunakan least squares
23    alpha = linalg.lstsq(A, b_v)[0] # koefisien vertical
24    beta = linalg.lstsq(A, b_h)[0] # koefisien horizontal

```

**Gambar 9.** Source code sistem persamaan linier  
(Sumber: Dokumentasi Pribadi)

### D. Validasi Model

Setelah model matematis selesai dikembangkan, validasi dilakukan untuk mengukur sejauh mana model mampu merepresentasikan pola *recoil* secara akurat. Validasi model dilakukan dengan membandingkan hasil prediksi model dengan data aktual menggunakan dataset uji. Tujuan utama dari proses ini adalah untuk mengevaluasi tingkat galat dan kekonsistenan model melalui metrik-metrik evaluasi yang sesuai. Metrik evaluasi yang digunakan dalam validasi adalah sebagai berikut:

### 1. Root Mean Squared Error (RMSE)

RMSE digunakan untuk mengukur deviasi rata-rata kuadrat antara data aktual dan prediksi model. Metrik ini memberikan indikasi seberapa jauh prediksi model dari nilai aktual, dengan memberikan penalti lebih besar untuk kesalahan yang lebih besar. RMSE dihitung dengan formula:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2}$$

dimana  $\hat{y}_i$  adalah prediksi model,  $y_i$  adalah nilai aktual, dan  $N$  adalah jumlah data.

### 2. R-squared ( $R^2$ )

$R^2$  digunakan untuk mengukur seberapa baik model dapat menjelaskan variabilitas data. Nilai  $R^2$  berkisar antara 0 hingga 1, dimana nilai mendekati 1 menunjukkan bahwa model mampu menjelaskan sebagian besar variabilitas data.

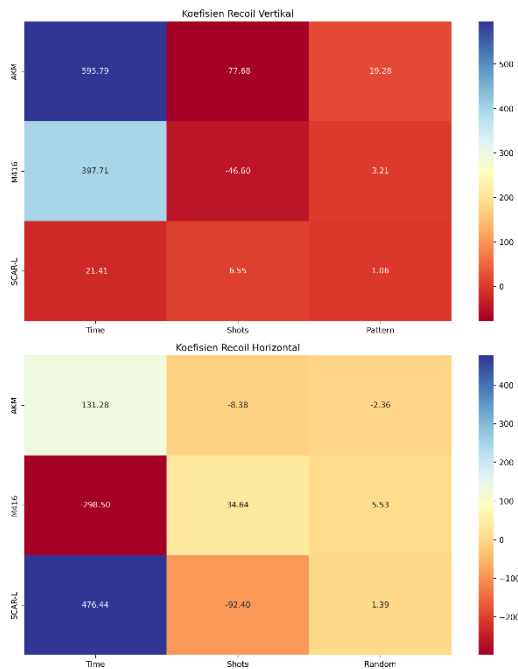
### 3. Mean Absolute Error (MAE)

MAE mengukur rata-rata kesalahan absolut antara prediksi dan data aktual, memberikan gambaran seberapa besar galat rata-rata tanpa memperhatikan arah galat.

## E. Visualisasi

Visualisasi digunakan untuk membantu memahami pola *recoil* dan validasi model secara lebih intuitif. Berikut adalah hasil visualisasi utama yang dihasilkan dari analisis:

### 1. Visualisasi Koefisien Model



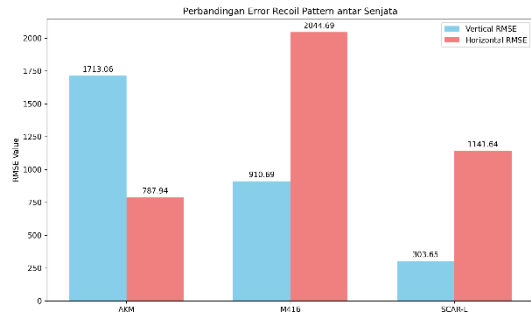
**Gambar 10.** Koefisien Recoil Vertikal dan Horizontal untuk Senjata AKM, M416, dan SCAR-L

(Sumber: Dokumentasi Pribadi Hasil Analisis Data Model)

Gambar diatas menunjukkan *heatmap* koefisien untuk dimensi vertikal dan horizontal. Heatmap ini memberikan gambaran distribusi nilai koefisien untuk variabel waktu ( $t$ ),

jumlah peluru ( $n$ ), dan parameter pola *recoil* ( $p$ ) serta faktor *random* ( $r$ ). Senjata SCAR-L, yang dikenal memiliki *recoil* paling stabil, menunjukkan koefisien vertikal dan horizontal yang lebih kecil dibandingkan senjata lain, terutama pada variabel waktu dan indeks peluru.

### 2. Visualisasi Validasi Model



**Gambar 11.** Perbandingan RMSE untuk Dimensi Vertikal dan Horizontal

(Sumber: Dokumentasi Pribadi Hasil Validasi Model Recoil Pattern)

Gambar 11 memberikan perbandingan nilai RMSE (Root Mean Squared Error) antara prediksi model dan data aktual untuk dimensi vertikal dan horizontal. Hasil ini menunjukkan bahwa model memiliki tingkat *error* yang berbeda pada setiap senjata. SCAR-L memiliki RMSE vertikal terendah (303.65) yang mengindikasikan bahwa model mampu memprediksi pola *recoil* dengan lebih akurat pada senjata ini. Sebaliknya, AKM memiliki RMSE vertikal tertinggi (1713.06), menunjukkan pola *recoil* yang lebih sulit diprediksi.

## IV. EKSPERIMEN

Kasus ideal untuk menunjukkan performa model adalah dengan memprediksi pola *recoil* pada data pengujian menggunakan senjata AKM, M416, dan SCAR-L. Senjata-senjata ini dipilih karena mewakili spektrum *recoil* yang berbeda, yaitu *recoil* tinggi (AKM), medium (M416), dan rendah (SCAR-L). Eksperimen dilakukan berdasarkan data yang telah dikumpulkan dari mode pelatihan PUBG Mobile, dengan pengaturan seperti yang dijelaskan pada bab sebelumnya.

Pengujian pertama dilakukan pada senjata AKM untuk memprediksi pola *recoil* pada dimensi vertikal dan horizontal. AKM memiliki spesifikasi *recoil pattern* sebesar 76, *random factor* (laju tembak) 84, *damage* 48, *reload speed* 76, dan jangkauan 52, yang menjadikannya salah satu senjata dengan *recoil* tertinggi di kelas *Assault Rifle*. Dari hasil analisis model, koefisien vertikal ( $\alpha$ ) menunjukkan bahwa waktu ( $t$ ) memiliki pengaruh positif yang signifikan terhadap pola *recoil* vertikal dengan nilai  $\alpha_1 = 595.79$ , sedangkan jumlah peluru ( $n$ ) memiliki pengaruh negatif dengan  $\alpha_2 = -77.68$ . Parameter pola *recoil* ( $p$ ) menunjukkan pengaruh yang lebih kecil tetapi tetap positif, dengan  $\alpha_3 = 19.28$ . Pada dimensi horizontal ( $\beta$ ), waktu ( $t$ ) memiliki pengaruh positif dengan  $\beta_1 = 131.28$ , sementara jumlah peluru ( $n$ ) memberikan kontribusi negatif dengan  $\beta_2 = -8.38$ , dan *random factor* ( $r$ ) juga menunjukkan dampak negatif dengan  $\beta_3 = -2.36$ .

Kesalahan rata-rata (RMSE) didapat sebesar 1713.06 untuk dimensi vertikal dan 787.94 untuk dimensi horizontal. RMSE vertikal yang tinggi menunjukkan bahwa model mengalami kesulitan dalam memprediksi pola *recoil* vertikal pada AKM, yang kemungkinan disebabkan oleh *recoil* AKM yang sangat dinamis dan tidak stabil. Namun, pada dimensi horizontal, RMSE yang lebih rendah menunjukkan bahwa model lebih mampu memprediksi distribusi *recoil* horizontal senjata ini. Hasil ini mengindikasikan bahwa model berhasil menangkap pola umum *recoil* AKM.

```

Analyzing AKM recoil pattern...

Koefisien untuk AKM:
Spesifikasi Senjata:
Recoil Pattern: 76
Random Factor (Laju Tembak): 84
Damage: 48
Reload Speed: 76
Jangkauan: 52

Hasil Analisis:
Vertical (α):
α1 (time): 595.7897
α2 (shots): -77.6807
α3 (pattern): 19.2800

Horizontal (β):
β1 (time): 131.2847
β2 (shots): -8.3820
β3 (random): -2.3566

RMSE (Root Mean Square Error):
Vertical: 1713.0550
Horizontal: 787.9401

```

**Gambar 12.** Hasil Analisis Pola Recoil AKM  
(Sumber: Dokumentasi Pribadi Hasil recoil\_model.py)

Pengujian selanjutnya dilakukan pada senjata M416 untuk memprediksi pola *recoil* pada dimensi vertikal dan horizontal. M416 memiliki spesifikasi *recoil pattern* sebesar 76, random factor 85, damage 41, kecepatan reload 76, dan jangkauan 50. Dari hasil analisis model, koefisien vertikal ( $\alpha$ ) menunjukkan bahwa waktu ( $t$ ) memiliki pengaruh positif terhadap pola *recoil* vertikal dengan nilai  $\alpha_1 = 397.71$ . Jumlah peluru ( $n$ ) memiliki pengaruh negatif dengan  $\alpha_2 = -46.60$ , sementara parameter pola *recoil* ( $p$ ) menunjukkan kontribusi yang lebih kecil tetapi tetap positif, dengan  $\alpha_3 = 3.21$ . Pada dimensi horizontal ( $\beta$ ), waktu ( $t$ ) memberikan pengaruh negatif yang signifikan dengan  $\beta_1 = -298.50$ , sedangkan jumlah peluru ( $n$ ) memberikan pengaruh positif dengan  $\beta_2 = 34.64$ , dan random factor ( $r$ ) juga memberikan kontribusi positif sebesar  $\beta_3 = 5.53$ .

Model menunjukkan kesalahan rata-rata (RMSE) sebesar 910.69 untuk dimensi vertikal dan 2044.69 untuk dimensi horizontal. RMSE vertikal yang lebih rendah dibandingkan dengan senjata AKM menunjukkan bahwa model lebih mampu memprediksi pola *recoil* vertikal pada M416. Hasil ini menunjukkan bahwa model dapat menangkap pola *recoil* vertikal pada M416 dengan akurasi yang lebih baik dibandingkan dimensi horizontal. Performa ini mengindikasikan bahwa *recoil* vertikal M416 memiliki pola yang lebih terstruktur dibandingkan recoil horizontal, yang memerlukan analisis lebih lanjut untuk meningkatkan akurasi model.

```

Analyzing M416 recoil pattern...

Koefisien untuk M416:
Spesifikasi Senjata:
Recoil Pattern: 76
Random Factor (Laju Tembak): 85
Damage: 41
Reload Speed: 76
Jangkauan: 50

Hasil Analisis:
Vertical (α):
α1 (time): 397.7071
α2 (shots): -46.5955
α3 (pattern): 3.2092

Horizontal (β):
β1 (time): -298.5023
β2 (shots): 34.6437
β3 (random): 5.5336

RMSE (Root Mean Square Error):
Vertical: 910.6941
Horizontal: 2044.6866

```

**Gambar 13.** Hasil Analisis Pola Recoil M416  
(Sumber: Dokumentasi Pribadi Hasil recoil\_model.py)

Pengujian terakhir dilakukan pada senjata SCAR-L untuk memprediksi pola *recoil* pada dimensi vertikal dan horizontal. SCAR-L memiliki spesifikasi *recoil pattern* sebesar 74, laju tembak 85, damage 41, reload speed 75, dan jangkauan 53, yang menjadikannya salah satu senjata dengan *recoil* paling stabil di Assault Rifle. Dari hasil analisis model, koefisien vertikal ( $\alpha$ ) menunjukkan bahwa waktu ( $t$ ) memiliki pengaruh negatif kecil terhadap pola *recoil* vertikal dengan nilai  $\alpha_1 = -21.41$ , sedangkan indeks peluru ( $n$ ) memiliki pengaruh positif sebesar  $\alpha_2 = 6.55$ . Parameter pola *recoil* ( $p$ ) memberikan kontribusi yang sangat kecil tetapi tetap positif, dengan  $\alpha_3 = 1.06$ . Pada dimensi horizontal ( $\beta$ ), waktu ( $t$ ) menunjukkan pengaruh positif yang signifikan dengan  $\beta_1 = 476.44$ , sementara jumlah peluru ( $n$ ) memberikan kontribusi negatif dengan  $\beta_2 = -92.40$ , dan random factor ( $r$ ) menunjukkan dampak positif yang kecil dengan  $\beta_3 = 1.39$ .

Hal tersebut memberikan kesalahan rata-rata (RMSE) sebesar 303.65 untuk sumbu vertikal dan 1141.64 untuk sumbu horizontal. RMSE vertikal yang jauh lebih rendah dibandingkan dengan AKM dan M416 menunjukkan bahwa model sangat efektif dalam memprediksi pola *recoil* vertikal pada SCAR-L. Pada dimensi horizontal, meskipun RMSE masih cukup besar, nilai tersebut tetap lebih rendah dibandingkan M416, yang menunjukkan bahwa distribusi *recoil* horizontal SCAR-L lebih stabil. Hasil ini mengindikasikan bahwa model berhasil menangkap pola *recoil* SCAR-L dengan baik, terutama pada dimensi vertikal. Stabilitas *recoil* yang lebih baik pada senjata ini mendukung performa model, sehingga akurasi prediksi meningkat dibandingkan senjata lain dengan *recoil* lebih dinamis seperti AKM dan M416. Hal ini menunjukkan bahwa SCAR-L memiliki pola *recoil* yang lebih terprediksi dibandingkan senjata lainnya.

```
Analyzing SCAR-L recoil pattern...

Koefisien untuk SCAR-L:
Spesifikasi Senjata:
Recoil Pattern: 74
Random Factor (Laju Tembak): 85
Damage: 41
Reload Speed: 75
Jangkauan: 53

Hasil Analisis:
Vertical ( $\alpha$ ):
 $\alpha_1$  (time): -21.4130
 $\alpha_2$  (shots): 6.5493
 $\alpha_3$  (pattern): 1.0589

Horizontal ( $\beta$ ):
 $\beta_1$  (time): 476.4372
 $\beta_2$  (shots): -92.4005
 $\beta_3$  (random): 1.3949

RMSE (Root Mean Square Error):
Vertical: 303.6510
Horizontal: 1141.6436
```

**Gambar 14.** Hasil Analisis Pola Recoil SCAR-L  
(Sumber: Dokumentasi Pribadi Hasil recoil\_model.py)

## V. KESIMPULAN

Model sistem persamaan linear yang dikembangkan mampu memberikan wawasan mendalam tentang pola *recoil* tiga senjata, yakni AKM, M416, dan SCAR-L. Hasil analisis menunjukkan bahwa SCAR-L memiliki stabilitas *recoil* terbaik, dengan RMSE terendah pada dimensi vertikal dan horizontal. M416 menawarkan keseimbangan dengan *recoil* yang lebih moderat, sementara AKM, dengan *recoil* tinggi, menunjukkan tantangan terbesar dalam prediksi pola *recoil*. Pemilihan senjata sebaiknya disesuaikan dengan gaya bermain dan kemampuan mengontrol *recoil*. Untuk meningkatkan akurasi, pendekatan non-linear dapat dieksplorasi lebih lanjut.

## VI. PENUTUP

Terima kasih kepada Allah SWT karena atas nikmat dan rahmat-Nya, penulis dapat menyelesaikan makalah berjudul "Implementasi Sistem Persamaan Linear untuk Prediksi Pola Recoil Senjata dalam PUBG Mobile" dengan baik. Selain itu, tidak lupa juga ucapan terima kasih kepada dosen mata kuliah Aljabar Linear dan Geometri, Ir. Rila Mandala, M.Eng., Ph.D., Dr. Ir. Rinaldi Munir, M. T., Dr. Judhi Santoso, M.Sc., dan Arrival Dwi Sentosa, S.Kom., M.T. yang telah membimbing penulis selama berkuliah di mata kuliah ini. Penulis juga mengucapkan terima kasih kepada seluruh sumber yang dijadikan referensi pada makalah ini.

## REFERENSI

- [1] PlayStore, "PUBG MOBILE," [Online]. Available: <https://play.google.com/store/apps/details?id=com.tencent.ig>. [Accessed 19 December 2024].
- [2] PUBG Mobile, "PUBG Mobile Wiki," [Online]. Available: <https://www.pubgmobile.com/id/event/PUBGMOBILE-WIKI/index.shtml>. [Accessed 25 December 2024].
- [3] M. Dr. Ir. Rinaldi, "Sistem Persamaan Linier," 2023. [Online]. Available: <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/Algeo-03-Sistem-Persamaan-Linier-2023.pdf>. [Accessed 20 December 2024].
- [4] M. Dr. Ir. Rinaldi, "Tiga Kemungkinan Solusi SPL," 2023. [Online]. Available: <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/Algeo-04-Tiga-Kemungkinan-Solusi-SPL-2023.pdf>. [Accessed 20 December 2024].

- [5] S. Glen, "Least Squares Regression Line," StatisticsHowTo.com, [Online]. Available: <https://www.statisticshowto.com/probability-and-statistics/regression-analysis/find-a-linear-regression-equation/>. [Accessed 23 December 2024].
- [6] OpenCV Team, "OpenCV: Video Analysis," [Online]. Available: [https://docs.opencv.org/4.x/d4/dee/tutorial\\_optical\\_flow.html](https://docs.opencv.org/4.x/d4/dee/tutorial_optical_flow.html). [Accessed 20 December 2024].

## TAUTAN KODE PROGRAM

[https://github.com/danenftyessir/PUBGM\\_Recoil\\_Analysis.git](https://github.com/danenftyessir/PUBGM_Recoil_Analysis.git)

## PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Jakarta, 27 Desember 2024



Danendra Shafi Athallah 13523136